Wentworth Institute of Technology
COMP1000 – Computer Science I
Fall 2017, Kreimendahl

Programming Assignment 6

In this assignment you are to write a program to solve the following problem. As with all assignments, remember the following submission steps:

- Make sure your code passes at least all the provided JUnit tests
- Create and test Javadoc code documentation
- Save, commit, and push all code changes
- Confirm the latest code is visible via the "Files" section of your repository website
- Confirm that the repository is private, and that the instructor has Developer access

Not that for this problem, some JUnit tests that will be used for grading have not been provided as a part of the starter code. It is your responsibility to thoroughly test your code.

Problem a (`PA6a.java`)

Write a program to generate the entire calendar for one year. The program must get two values from the user: (1) the year and (2) the day of the week for January 1st of that year. The year, which should be positive, is needed to check for and handle leap years[1]. The day of the week for January 1st is needed so that you know where to start the calendar. The user should enter 0 for Sunday, 1 for Monday, … or 6 for Saturday. As always, you need to validate the user's input. To actually print the calendar, you must use a single method that prints out the calendar for one month and then call this function 12 times from main(), once for each month in the year. To check for a leap year you will need to write another method that takes the year as a parameter and returns true if it's a leap year, or false otherwise. Stubs (i.e. method signatures without any code) for both of these methods have been provided for you.

The calendar should be printed in the form below. In this example, January starts on a Saturday (day 6). Note that February starts on a Tuesday in this example because January ended on a Monday.

```
January
                    1
  2  3  4  5  6  7  8
  9 10 11 12 13 14 15
 16 17 18 19 20 21 22
 23 24 25 26 27 28 29
 30 31

February
        1  2  3  4  5
  6  7  8  9 10 11 12
…
```

[1] See https://en.wikipedia.org/wiki/Leap_year - Algorithm

This is a very complex program with many pieces, and there are many different ways to approach it. Before writing any code, take some time to think through the problem. The most difficult piece is to understand how to write a single method that can print out the calendar for any month. Here is a suggestion for how to proceed:

1. Try to write the isLeapYear method – this is self-contained and there is a unit test to help you determine if you understand the logic.

2. Now focus on the printMonth method – there is another unit test to help you make sure you understand the logic here. Start with printing the name of the month (supplied as a parameter) and trailing new lines. Then try getting the correct number of days printed (also supplied as a parameter) without worrying about which day the month starts on. Then focus on getting the numbers to line up (hint: how are you going to deal with a single digit vs. double digits), then on starting a new line after each Saturday. Now make sure your return value is correct – it should be one day of the week later than the last printed day (with special handling if the last day is Saturday). Finally, print "blank" entries for each day until the first of the month. For example, if January 1st is a Wednesday, you need to print 3 blank entries.

3. Next, shift your focus to the main() method. Start by getting input from the user and validating them (there are unit tests focused on these, as well as error messages provided for you).

4. Now, assuming the inputs are supplied correctly, call your methods to print out each month (i.e. 12 calls to printMonth). Note that you will need to use the return value from this method to decide where to start each subsequent month.

5. Finally, add correct support for leap years – think about how to use your isLeapYear method to correctly output February.

You have been supplied JUnit tests for the two methods, validating the inputs in main(), as well as the output for the entire calendar for the year 2015.